Nonlinear independent component analysis: A principled framework for unsupervised deep learning

Aapo Hyvärinen

[Now:] Parietal Team, INRIA-Saclay, France
[Earlier:] Gatsby Unit, University College London, UK
[Always:] Dept of Computer Science, University of Helsinki, Finland
[Kind of:] CIFAR
Abstract

▶ Short critical introduction to deep learning
▶ Importance of Big Data
Abstract

- Short critical introduction to deep learning
  - Importance of Big Data
- Importance of unsupervised learning
Abstract

- Short critical introduction to deep learning
  - Importance of Big Data
- Importance of unsupervised learning
- Disentanglement methods try to find independent factors
Abstract

- Short critical introduction to deep learning
  - Importance of Big Data
- Importance of unsupervised learning
- Disentanglement methods try to find independent factors
- In linear case, independent component analysis (ICA) successful, can we extend to a nonlinear method?
Abstract

- Short critical introduction to deep learning
  - Importance of Big Data
- Importance of unsupervised learning
- Disentanglement methods try to find independent factors
- In linear case, independent component analysis (ICA) successful, can we extend to a nonlinear method?
- Problem: Nonlinear ICA fundamentally ill-defined
Abstract

- Short critical introduction to deep learning
  - Importance of Big Data
- Importance of unsupervised learning
- Disentanglement methods try to find independent factors
- In linear case, independent component analysis (ICA) successful, can we extend to a nonlinear method?
- Problem: Nonlinear ICA fundamentally ill-defined
- Solution 1: use temporal structure in time series, in a self-supervised fashion
Abstract

- Short critical introduction to deep learning
  - Importance of Big Data
- Importance of unsupervised learning
- Disentanglement methods try to find independent factors
- In linear case, independent component analysis (ICA) successful, can we extend to a nonlinear method?
- Problem: Nonlinear ICA fundamentally ill-defined
- Solution 1: use temporal structure in time series, in a self-supervised fashion
- Solution 2: use an extra auxiliary variable in a VAE framework
Success of Artificial Intelligence

- Autonomous vehicles, machine translation, game playing, search engines, recommendation machine, etc.

- Most modern applications based on deep learning
Neural networks

- Layers of “neurons” repeating linear transformations and simple nonlinearities $f$
  \[ x_i(L + 1) = f \left( \sum_j w_{ij}(L)x_j(L) \right), \]  where $L$ is layer

  with e.g. $f(x) = \max(0, x)$

- Can approximate “any” nonlinear input-output mappings

- Learns by nonlinear regression (e.g. least-squares)
Deep learning

- Deep Learning = learning in neural network with many layers
- With enough data, can learn any input-output relationship: image-category / past-present / friends - political views
- Present boom started by Krizhevsky, Sutskever, Hinton, 2012: Superior recognition success of objects in images
Characteristics of deep learning

- **Nonlinearity**: E.g. recognition of a cat is highly nonlinear
  - A linear model would use a single prototype
    - But locations, sizes, viewpoints highly variable
Characteristics of deep learning

- **Nonlinearity:** E.g. recognition of a cat is highly nonlinear
  - A linear model would use a single prototype
  - But locations, sizes, viewpoints highly variable

- Needs **big data**: E.g. millions of images from the Internet
  - Because general nonlinear functions have many parameters
Characteristics of deep learning

▶ **Nonlinearity**: E.g. recognition of a cat is highly nonlinear
  ▶ A linear model would use a single prototype
    But locations, sizes, viewpoints highly variable

▶ Needs **big data**: E.g. millions of images from the Internet
  ▶ Because general nonlinear functions have many parameters

▶ Needs **big computers**: Graphics Processing Units (GPU)
  ▶ Obvious consequence of need for big data, and nonlinearities
Characteristics of deep learning

- **Nonlinearity**: E.g. recognition of a cat is highly nonlinear
  - A linear model would use a single prototype
    But locations, sizes, viewpoints highly variable

- Needs **big data**: E.g. millions of images from the Internet
  - Because general nonlinear functions have many parameters

- Needs **big computers**: Graphics Processing Units (GPU)
  - Obvious consequence of need for big data, and nonlinearities

- Most **theory quite old**: Nonlinear (logistic) regression
  - But earlier we didn’t have enough data and “compute”
Success stories in deep learning need category labels

- Is it a cat or a dog? Liked or not liked?
Importance unsupervised learning

- Success stories in deep learning need category labels
  - Is it a cat or a dog? Liked or not liked?
- Problem: labels may be
  - Difficult to obtain
  - Unrealistic in neural modelling
  - Ambiguous
Importance unsupervised learning

- Success stories in deep learning need category labels
  - Is it a cat or a dog? Liked or not liked?
- Problem: labels may be
  - Difficult to obtain
  - Unrealistic in neural modelling
  - Ambiguous
Importance unsupervised learning

- Success stories in deep learning need category labels
  - Is it a cat or a dog? Liked or not liked?
- Problem: labels may be
  - Difficult to obtain
  - Unrealistic in neural modelling
  - Ambiguous

- Unsupervised learning:
  - we only observe a data vector \( \mathbf{x} \), no label or target \( \mathbf{y} \)
  - E.g. photographs with no labels
- Very difficult, largely unsolved problem
ICA as principled unsupervised learning

Linear independent component analysis (ICA)

\[ x_i(t) = \sum_{j=1}^{n} a_{ij} s_j(t) \quad \text{for all } i, j = 1 \ldots n \] (2)

\[ x_i(t) \] is the \( i \)-th observed signal at sample point \( t \) (possibly time)

\( a_{ij} \) are constant parameters describing “mixing”

Assuming independent, non-Gaussian latent “sources” \( s_j \)
ICA as principled unsupervised learning

- Linear independent component analysis (ICA)

\[ x_i(t) = \sum_{j=1}^{n} a_{ij} s_j(t) \quad \text{for all } i, j = 1 \ldots n \] (2)

- \( x_i(t) \) is \( i \)-th observed signal at sample point \( t \) (possibly time)
- \( a_{ij} \) constant parameters describing “mixing”
- Assuming independent, non-Gaussian latent “sources” \( s_j \)
- ICA is identifiable, i.e. well-defined: (Darmois-Skitovich \sim 1950; Comon, 1994)
  - Observing only \( x_i \) we can recover both \( a_{ij} \) and \( s_j \)
  - I.e. original sources can be recovered
  - As opposed to PCA, factor analysis
Unsupervised learning can have different goals

1) Accurate model of data distribution?
   ▶ E.g. Variational Autoencoders are good

2) Sampling points from data distribution?
   ▶ E.g. Generative Adversarial Networks are good

3) Useful features for supervised learning?
   ▶ Many methods, “Representation learning”

4) Reveal underlying structure in data, disentangle latent quantities?
   ▶ Independent Component Analysis! (this talk)

▶ These goals are orthogonal, even contradictory!
▶ Probably, no method can accomplish all (Cf. Theis et al 2015)
▶ In unsupervised learning research, must specify actual goal
Unsupervised learning can have different goals

1) Accurate model of data distribution?
   ▶ E.g. Variational Autoencoders are good
Unsupervised learning can have different goals

1) Accurate model of data distribution?
   - E.g. Variational Autoencoders are good

2) Sampling points from data distribution?
   - E.g. Generative Adversarial Networks are good
Unsupervised learning can have different goals

1) Accurate model of data distribution?
   ▶ E.g. Variational Autoencoders are good
2) Sampling points from data distribution?
   ▶ E.g. Generative Adversarial Networks are good
3) Useful features for supervised learning?
   ▶ Many methods, “Representation learning”
Unsupervised learning can have different goals

1) Accurate model of data distribution?
   ▶ E.g. Variational Autoencoders are good

2) Sampling points from data distribution?
   ▶ E.g. Generative Adversarial Networks are good

3) Useful features for supervised learning?
   ▶ Many methods, “Representation learning”

4) Reveal underlying structure in data, disentangle latent quantities?
   ▶ Independent Component Analysis! (this talk)
Unsupervised learning can have different goals

1) Accurate model of data distribution?
   ▶ E.g. Variational Autoencoders are good

2) Sampling points from data distribution?
   ▶ E.g. Generative Adversarial Networks are good

3) Useful features for supervised learning?
   ▶ Many methods, “Representation learning”

4) Reveal underlying structure in data, disentangle latent quantities?
   ▶ Independent Component Analysis! (this talk)
   ▶ These goals are orthogonal, even contradictory!
   ▶ Probably, no method can accomplish all (Cf. Theis et al 2015)
Unsupervised learning can have different goals

1) Accurate model of data distribution?
   ▶ E.g. Variational Autoencoders are good

2) Sampling points from data distribution?
   ▶ E.g. Generative Adversarial Networks are good

3) Useful features for supervised learning?
   ▶ Many methods, “Representation learning”

4) Reveal underlying structure in data, disentangle latent quantities?
   ▶ Independent Component Analysis! (this talk)
   ▶ These goals are orthogonal, even contradictory!
     ▶ Probably, no method can accomplish all (Cf. Theis et al 2015)
   ▶ In unsupervised learning research, must specify actual goal
Identifiability means ICA does blind source separation

Observed signals:

Principal components:

Independent components are original sources:
Example of ICA: Brain source separation

(Hyvärinen, Ramkumar, Parkkonen, Hari, 2010)
Example of ICA: Image features

(Olshausen and Field, 1996; Bell and Sejnowski, 1997)

Features similar to wavelets, Gabor functions, simple cells.
Nonlinear ICA is an unsolved problem

- Extend ICA to nonlinear case to get general disentanglement?
- Unfortunately, “basic” nonlinear ICA is not identifiable:
- If we define nonlinear ICA model simply as

\[ x_i(t) = f_i(s_1(t), \ldots, s_n(t)) \quad \text{for all } i, j = 1 \ldots n \]  

we cannot recover original sources (Darmois, 1952; Hyvärinen & Pajunen, 1999)
Darmois construction

- Darmois (1952) showed impossibility of nonlinear ICA:
- For any $x_1, x_2$, can always construct $y = g(x_1, x_2)$
  independent of $x_1$ as

$$g(\xi_1, \xi_2) = P(x_2 < \xi_2 | x_1 = \xi_1)$$  \hspace{1cm} (4)
Darmois construction

- Darmois (1952) showed impossibility of nonlinear ICA:
- For any $x_1, x_2$, can always construct $y = g(x_1, x_2)$ independent of $x_1$ as
  \[
g(\xi_1, \xi_2) = P(x_2 < \xi_2|x_1 = \xi_1)
\]

- Independence alone too weak for identifiability:
  We could take $x_1$ as independent component which is absurd
- Maximizing non-Gaussianity of components equally absurd:
  Scalar transform $h(x_1)$ can give any distribution
Temporal structure helps in nonlinear ICA

- Two kinds of temporal structure:
  - Autocorrelations (Harmeling et al 2003)
  - Nonstationarity (Hyvärinen and Morioka, NIPS2016)

- Now, identifiability of nonlinear ICA can be proven (Sprekeler et al, 2014; Hyvärinen and Morioka, NIPS2016 & AISTATS2017):
  Can find original sources!
Trick: “Self-supervised” learning

- Supervised learning: we have
  - “input” $x$, e.g. images / brain signals
  - “output” $y$, e.g. content (cat or dog) / experimental condition
Trick: “Self-supervised” learning

- **Supervised learning**: we have
  - “input” $x$, e.g. images / brain signals
  - “output” $y$, e.g. content (cat or dog) / experimental condition

- **Unsupervised learning**: we have
  - only “input” $x$
Trick: “Self-supervised” learning

- **Supervised learning**: we have
  - “input” $x$, e.g. images / brain signals
  - “output” $y$, e.g. content (cat or dog) / experimental condition

- **Unsupervised learning**: we have
  - only “input” $x$

- **Self-supervised** learning: we have
  - only “input” $x$
  - *but we invent* $y$ somehow, e.g. by creating corrupted data, and use supervised algorithms
Deep Learning
Independent component analysis
Nonlinear ICA
Connection to VAE’s

ICA as principled unsupervised learning
Difficulty of nonlinear ICA

Trick: “Self-supervised” learning

- **Supervised learning**: we have
  - “input” $x$, e.g. images / brain signals
  - “output” $y$, e.g. content (cat or dog) / experimental condition

- **Unsupervised learning**: we have
  - only “input” $x$

- **Self-supervised** learning: we have
  - only “input” $x$
  - *but we invent $y$* somehow, e.g. by creating corrupted data, and use supervised algorithms

- Numerous examples in computer vision:
  - Remove part of photograph, learn to predict missing part
    ($x$ is original data with part removed, $y$ is missing part)
Permutation-contrastive learning (Hyvärinen and Morioka 2017)

- Observe $n$-dim time series $\mathbf{x}(t)$
Permutation-contrastive learning (Hyvärinen and Morioka 2017)

- Observe $n$-dim time series $x(t)$
- Take short time windows as new data

$$y(t) = (x(t), x(t - 1))$$
Permutation-contrastive learning (Hyvärinen and Morioka 2017)

- Observe $n$-dim time series $\mathbf{x}(t)$
- Take short time windows as new data
  \[ \mathbf{y}(t) = (\mathbf{x}(t), \mathbf{x}(t-1)) \]
- Create randomly time-permuted data
  \[ \mathbf{y}^*(t) = (\mathbf{x}(t), \mathbf{x}(t^*)) \]

with $t^*$ a random time point.
Permutation-contrastive learning *(Hyvärinen and Morioka 2017)*

- Observe $n$-dim time series $\mathbf{x}(t)$
- Take short time windows as new data
  \[
  \mathbf{y}(t) = (\mathbf{x}(t), \mathbf{x}(t-1))
  \]
- Create randomly time-permuted data
  \[
  \mathbf{y}^*(t) = (\mathbf{x}(t), \mathbf{x}(t^*))
  \]
  with $t^*$ a random time point.
- Train NN to discriminate $\mathbf{y}$ from $\mathbf{y}^*$
- Could this really do Nonlinear ICA?
Theorem: PCL estimates nonlinear ICA with time dependencies

- Assume data follows nonlinear ICA model $x(t) = f(s(t))$ with
  - smooth, invertible nonlinear mixing $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$
  - independent sources $s_i(t)$
    - temporally dependent (strongly enough), stationary
    - non-Gaussian (strongly enough)
Theorem: PCL estimates nonlinear ICA with time dependencies

- Assume data follows nonlinear ICA model $x(t) = f(s(t))$ with
  - smooth, invertible nonlinear mixing $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$
  - independent sources $s_i(t)$
    - temporally dependent (strongly enough), stationary
    - non-Gaussian (strongly enough)
- Then, PCL demixes nonlinear ICA: hidden units give $s_i(t)$
  - A constructive proof of identifiability
Theorem: PCL estimates nonlinear ICA with time dependencies

- Assume data follows nonlinear ICA model $x(t) = f(s(t))$ with
  - smooth, invertible nonlinear mixing $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$
  - independent sources $s_i(t)$
    - temporally dependent (strongly enough), stationary
    - non-Gaussian (strongly enough)
- Then, PCL demixes nonlinear ICA: hidden units give $s_i(t)$
  - A constructive proof of identifiability
- For Gaussian sources, demixes up to linear mixing
Illustration of demixing capability

- AR Model with Laplacian innovations, $n = 2$
  \[
  \log p(s(t)|s(t-1)) = -|s(t) - \rho s(t-1)|
  \]
- Nonlinearity is MLP. Mixing: leaky ReLU’s; Demixing: maxout

Sources ($s$)  
Mixtures ($x$)  
Estimates by kTDSEP (Harmeling et al 2003)  
Estimates by our PCL
Time-contrastive learning: (Hyvärinen and Morioka 2016)

- Observe $n$-dim time series $\mathbf{x}(t)$
Time-contrastive learning: (Hyvärinen and Morioka 2016)

- Observe $n$-dim time series $\mathbf{x}(t)$
- Divide $\mathbf{x}(t)$ into $T$ segments (e.g. bins with equal sizes)
Time-contrastive learning: (Hyvärinen and Morioka 2016)

- Observe $n$-dim time series $\mathbf{x}(t)$
- Divide $\mathbf{x}(t)$ into $T$ segments (e.g. bins with equal sizes)
- Train MLP to tell which segment a single data point comes from
  - Number of classes is $T$, labels given by index of segment
  - Multinomial logistic regression

Feature extractor: $\mathbf{h}(\mathbf{x}; \theta)$

Multinomial logistic regression: $\mathbf{W}, \mathbf{b}$
Time-contrastive learning: (Hyvärinen and Morioka 2016)

- Observe $n$-dim time series $\mathbf{x}(t)$
- Divide $\mathbf{x}(t)$ into $T$ segments (e.g. bins with equal sizes)
- Train MLP to tell which segment a single data point comes from
  - Number of classes is $T$, labels given by index of segment
  - Multinomial logistic regression
- In hidden layer $\mathbf{h}$, NN should learn to represent nonstationarity (≡ differences between segments)
- Nonlinear ICA for nonstationary data!
Experiments on MEG

- Sources estimated from resting data (no stimulation)
- a) Validation by classifying another data set with four stimulation modalities: visual, auditory, tactile, rest.
  - Trained a linear SVM on estimated sources
  - Number of layers in MLP ranging from 1 to 4
- b) Attempt to visualize nonlinear processing
Auxiliary variables: Alternative to temporal structure
(Arandjelovic & Zisserman, 2017; Hyvärinen et al, 2019)

Look at correlations of video (main data) and audio (aux var)

Figure 3. **Learned visual concepts.** Each column shows five images that most activate a particular unit of the 512 in pool4 for the vision...
Deep Learning
Independent component analysis
Nonlinear ICA
Connection to VAE’s

Deep Latent Variable Models and VAE’s

- General framework with observed data vector $x$ and latent $z$:
  \[
p(x, z) = p(x|z)p(z), \quad p(x) = \int p(x, z)dz
  \]
  where $\theta$ is a vector of parameters, e.g. in a neural network
- Posterior $p(x|z)$ could model nonlinear mixing

Variational autoencoders (VAE):
- Model:
  - Define prior so that $z$ white Gaussian (thus independent $z_i$)
  - Define posterior so that $x = f(z) + n$
- Estimation:
  - Approximative maximization of likelihood
  - Approximation is “variational lower bound”
  - Is such a model identifiable?

A. Hyvärinen Nonlinear ICA
Deep Latent Variable Models and VAE’s

- General framework with observed data vector $x$ and latent $z$:
  \[ p(x, z) = p(x|z)p(z), \quad p(x) = \int p(x, z)dz \]
  
  where $\theta$ is a vector of parameters, e.g. in a neural network

- Posterior $p(x|z)$ could model nonlinear mixing

- **Variational autoencoders (VAE):**
  - **Model:**
    - Define prior so that $z$ white Gaussian (thus independent $z_i$)
    - Define posterior so that $x = f(z) + n$
  - **Estimation:**
    - Approximative maximization of likelihood
    - Approximation is “variational lower bound”
Deep Latent Variable Models and VAE’s

- General framework with observed data vector $\mathbf{x}$ and latent $\mathbf{z}$:
  \[
p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z}), \quad p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{z})d\mathbf{z}
\]
  where $\theta$ is a vector of parameters, e.g. in a neural network

- Posterior $p(\mathbf{x}|\mathbf{z})$ could model nonlinear mixing

- **Variational autoencoders (VAE):**
  - **Model:**
    - Define prior so that $\mathbf{z}$ white Gaussian (thus independent $z_i$)
    - Define posterior so that $\mathbf{x} = f(\mathbf{z}) + \mathbf{n}$
  - **Estimation:**
    - Approximative maximization of likelihood
    - Approximation is “variational lower bound”

- Is such a model identifiable?
Identifiable VAE

▶ Original VAE is not identifiable:
  ▶ Latent variables usually white and Gaussian:
  ▶ Any orthogonal rotation is equivalent: $z' = Uz$ has exactly the same distribution.
Identifiable VAE

- Original VAE is not identifiable:
  - Latent variables usually white and Gaussian:
  - Any orthogonal rotation is equivalent: \( z' = Uz \) has exactly the same distribution.

- Our new iVAE (Khemakhem, Kingma, Hyvärinen, 2019):
  - Assume we also observe auxiliary variable \( u \), e.g. audio for video, segment label, history
  - General framework, not just time structure

- \( z_i \) conditionally independent given \( u \)
- Variant of our nonlinear ICA, hence identifiable
Application to causal analysis

- **Causal discovery**: learning causal structure without interventions
- We can use nonlinear ICA to find general non-linear causal relationships (Monti et al, UAI2019)
- Identifiability absolutely necessary

\[
S_1 : \quad X_1 = f_1(N_1) \\
S_2 : \quad X_2 = f_2(X_1, N_2)
\]
Conclusion

- Conditions for ordinary deep learning:
  - Big data, big computers, class labels (outputs)
Conclusion

- Conditions for ordinary deep learning:
  - Big data, big computers, class labels (outputs)
  - If no class labels: \textit{unsupervised} learning
Conclusion

- Conditions for ordinary deep learning:
  - Big data, big computers, class labels (outputs)
  - If no class labels: **unsupervised learning**
- Independent component analysis can be made nonlinear
  - Special assumptions needed for identifiability
Conclusion

- Conditions for ordinary deep learning:
  - Big data, big computers, class labels (outputs)
- If no class labels: unsupervised learning
- Independent component analysis can be made nonlinear
  - Special assumptions needed for identifiability
- Self-supervised methods are easy to implement
Conclusion

- Conditions for ordinary deep learning:
  - Big data, big computers, class labels (outputs)
- If no class labels: unsupervised learning
- Independent component analysis can be made nonlinear
  - Special assumptions needed for identifiability
- Self-supervised methods are easy to implement
- Connection to VAE’s can be made → iVAE
Conclusion

- Conditions for ordinary deep learning:
  - Big data, big computers, class labels (outputs)
- If no class labels: unsupervised learning
- Independent component analysis can be made nonlinear
  - Special assumptions needed for identifiability
- Self-supervised methods are easy to implement
- Connection to VAE’s can be made → iVAE
- Principled framework for “disentanglement”